AE/Tong

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application No.	10/607,158
Filing Date	June 25, 2003
First Named Inventor	Koichi Yamada
Art Unit	2113
Examiner Name	Amine Riad
Total Number of Pages in This Submission: 22	Attorney Docket Number: 42P15793

ENCLOSURES *(check all that apply)*

- [X] Fee Transmittal Form
  - [X] Fee Attached
- [ ] Amendment / Reply
  - [ ] After Final
  - [ ] Affidavits/declaration(s)
- [ ] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
  - [ ] PTO/SB/08
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts/ Incomplete Application
  - [ ] Basic Filing Fee
  - [ ] Declaration/POA
  - [ ] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s)
  - [ ] Landscape Table on CD
- [ ] After Allowance Communication to TC
- [ ] Appeal Communication to Board of Appeals and Interferences
- [X] Appeal Communication to TC (Appeal Notice, Brief, Reply Brief)
- [ ] Proprietary Information
- [ ] Status Letter
- [X] Other Enclosure(s) *(please identify below):*

--return receipt postcard

Remarks

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	Tong J. Lee, Reg. No. 48,582 BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
Signature	Tong Lee
Date	January 18, 2007

CERTIFICATE OF MAILING/TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Typed or printed name	**Amber D. Saunders**		
Signature	Amber D Saunders	Date	**January 18, 2007**

Based on PTO/SB/21 (09-04) as modified by Blakely, Sokoloff, Taylor & Zafman (ndc) 10/12/2006.
SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

FEE TRANSMITTAL for FY 2006

Patent fees are subject to annual revision.

☐ Applicant claims small entity status. See 37 CFR 1.27.

TOTAL AMOUNT OF PAYMENT	($)	500.00

Complete if Known	
Application Number	10/607,158
Filing Date	June 25, 2003
First Named Inventor	Koichi Yamada
Examiner Name	Amine Riad
Art Unit	2113
Attorney Docket No.	42P15793

METHOD OF PAYMENT *(check all that apply)*

☒ Check ☐ Credit card ☐ Money Order ☐ None ☐ Other (please identify): _____

☒ Deposit Account Deposit Account Number: 02-2666 Deposit Account Name: Blakely, Sokoloff, Taylor & Zafman LLP

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below
☐ Charge fee(s) indicated below, except for the filing fee
☒ Charge any additional fee(s) or underpayment of fee(s) under 37 CFR §§ 1.16, 1.17, 1.18 and 1.20.
☒ Credit any overpayments

FEE CALCULATION

1. EXTRA CLAIM FEES

			Extra Claims		Fee from below		Fee Paid
Total Claims	26	- 26* =	0	x	50.00	=	$0.00
Independent Claims	4	- 4* =	0	x	200.00	=	$0.00
Multiple Dependent						=	

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
1202	50	2202	25	Claims in excess of 20
1201	200	2201	100	Independent claims in excess of 3
1203	360	2203	180	Multiple Dependent claim, if not paid
1204	790	2204	395	**Reissue independent claims over original patent
1205	300	2205	150	**Reissue claims in excess of 20 and over original patent

**or number previously paid, if greater, For Reissues, see below

SUBTOTAL (1) ($) 0.00

2. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
1051	130	2051	65	Surcharge - late filing fee or oath	
1052	50	2052	25	Surcharge - late provisional filing fee or cover sheet.	
2053	130	2053	130	Non-English specification	
1251	120	2251	60	Extension for reply within first month	
1252	450	2252	225	Extension for reply within second month	
1253	1,020	2253	510	Extension for reply within third month	
1254	1,590	2254	795	Extension for reply within fourth month	
1255	2,160	2255	1,080	Extension for reply within fifth month	
1401	500	2401	250	Notice of Appeal	
1402	500	2402	250	Filing a brief in support of an appeal	500.00
1403	1,000	2403	500	Request for oral hearing	
1451	1,510	2451	1,510	Petition to institute a public use proceeding	
1460	130	2460	130	Petitions to the Commissioner	
1807	50	1807	50	Processing fee under 37 CFR 1.17(q)	
1806	180	1806	180	Submission of Information Disclosure Stmt	
1809	790	1809	395	Filing a submission after final rejection (37 CFR § 1.129(a))	
1810	790	2810	395	For each additional invention to be examined (37 CFR § 1.129(b))	
Other fee (specify)					

SUBTOTAL (2) ($) 500.00

SUBMITTED BY

Complete (if applicable)

Name (Print/Type)	Tong J. Lee	Registration No. (Attorney/Agent)	48,582	Telephone	(310) 207-3800
Signature				Date	01/18/07

Based on PTO/SB/17 (12-04) as modified by Blakely, Sokoloff, Taylor & Zafman (wlr) 12/15/2004.
SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

Attorney's Docket No. 042390.P15793

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application for:

Koichi Yamada

Serial No. 10/607,158

Filed: June 25, 2003

For: **IDENTIFYING AFFECTED PROGRAM THREADS AND ENABLING ERROR CONTAINMENT AND RECOVERY**

Examiner: Amine Riad

Art Unit: 2113

Confirmation No.: 5608

Assistant Commissioner for Patents
Board of Patent Appeals and Interferences
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Pursuant to 37 C.F.R. § 41.37, Appellant submits the following Appeal Brief for consideration by the Board of Patent Appeals and Interferences ("Board"). Appellant also submits herewith a check in the amount of $500.00 to cover the cost of filing this appeal brief, as set forth in 37 C.F.R. § 41.20(b)(2). Please charge any additional amounts due or credit any overpayment to Deposit Account No. 02-2666.

01/25/2007 CNEGA1 00000029 10607158
01 FC:1402 500.00 OP

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

Koichi Yamada, the party named in the caption, transferred his/her rights in the subject application "Identifying Affected Program Threads and Enabling Error Containment and Recovery" through an assignment recorded on reel/frame 014245/0095 on June 25, 2003 to Intel Corporation of Santa Clara, California. Thus, as the owner at the time the brief is being filed, Intel Corporation of Santa Clara, California is the real party in interest.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences that will directly affect, be directly affected by, or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-26 are pending in this application. All claims stand rejected. Appellant seeks review of claims 1-26.

IV. STATUS OF AMENDMENTS

No amendment has been submitted after the Final Office Action mailed August 22, 2006.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The claimed subject matter relates to the identification of an affected application program thread to enable error containment and recovery (paragraph 10). Four independent claims (claims 1, 8, 13, and 20) and twenty-two dependent claims (claims 2-7, 9-12, 14-19, and 21-26) are presented in this appeal.

According to the disclosed subject matter, an operating system receives machine error information of the operation mode of an offending application program thread, and terminates an affected application program thread if the thread is in the user operation mode (paragraph 11). An offending application program thread is the thread that issued an instruction causing a machine check abort ("MCA" or hardware error signal) from a hardware device. *id.* An affected application program thread is the thread in whose context the MCA was reported. *id.* The operating system distinguishes when to terminate the affected thread and let the other application programs continue, or when to shut down the entire software system. *id.* By identifying the operation mode of the affected thread, the operating system avoids always terminating the entire system after receiving a hardware data error (paragraph 35). This significantly increases the reliability and availability of a computer system (paragraph 46). In addition, the disclosed subject matter makes error recovery possible by the operating system without investing in expensive hardware support of the precise error reporting. *id.* This allows the operating system to implement error identification and recovery without extensive operating system changes. *id.*

Independent claim 1 is a method of terminating an affected application program thread (paragraph 11). The method comprises receiving an indication of a hardware error, the hardware error being associated with an application program thread (paragraph 24). If it is determined that the application program thread is in a user operation mode, the application program is terminated (paragraphs 30, 31, and 33).

Dependent claim 2 modifies the method of claim 1 to further include determining the hardware error is a memory read error, the memory read error being associated with the application program thread (paragraph 24).

Dependent claim 3 modifies the method of claim 2 to further include determining the memory read error is successfully contained (paragraph 28).

Dependent claim 4 modifies the method of claim 3 to further include receiving information of whether the memory read error is contained (paragraph 28).

Dependent claim 5 modifies the method of claim 2 to further include receiving information of whether the hardware error occurred on a memory read (paragraph 24).

Dependent claim 6 modifies the method of claim 1 to further include receiving information of a poisoned data address associated with the hardware error (paragraph 34).

Dependent claim 7 modifies the method of claim 1 to further include confirming one or more registers associated with the application program thread are consumed (paragraph 37).

Independent claim 8 is a system, which includes a processor to perform an instruction from an operating system (Figure 1, feature 105, and paragraphs 15 and 18). The system also includes a memory component to provide machine error information to the operating system (paragraph 24). The machine error information includes an operation mode of the affected application program (paragraphs 30 and 31). The operating system terminates the affected application program thread upon determining the affected application program to be within a user operation mode (paragraph 33).

Dependent claim 9 modifies the system of claim 8 to further include wherein the processor is to receive an instruction from the operating system to terminate the affected application program thread upon determining a memory read error has occurred (paragraph 24).

Dependent claim 10 modifies the system of claim 9 to further include wherein the processor is to receive an instruction from the operating system to terminate the affected application program thread upon determining the memory read error is contained (paragraph 28).

Dependent claim 11 modifies the system of claim 9 to further include wherein the operating system is to check to the machine error information message to determine whether the memory read error occurred (paragraph 24).

Dependent claim 12 modifies the system of claim 9 to further include wherein the operating system is to check the machine error information message to determine whether the memory read error is contained (paragraph 28).

Independent claim 13 is a machine-accessible medium that provides instructions that, if executed by a machine, will cause the machine to perform operations (paragraph 45). The operations comprise receiving an indication of a hardware error associated with an application program thread (paragraph 24), determining the application program thread to be in a user operation mode (paragraphs 30 and 31), and terminating the application program (paragraphs 33).

Dependent claims 14-19 modify claim 13 to further include the operations corresponding to the method of claims 2-7 (paragraphs 24, 28, 34, 37, and 45).

Independent claim 20 is system, which includes a means for receiving an indication of a hardware error associated with an application program thread (paragraph 24), a means for determining the application program thread to be in a user operation mode (paragraphs 30 and 31), and a means for terminating the application program (paragraphs 33).

Dependent claims 21-26 modify the system of claim 20 to further include the elements corresponding to the system of claims 2-7 (paragraphs 24, 28, 34, 37, and 45).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-5, 7-17, 19-24, and 26 stand rejected under 35 U.S.C. § 102(a) as being anticipated by U.S. Patent No. 6,938,254 issued to Mathur et al. ("Mathur").

Claims 6, 18 and 25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable in view of Mathur over U.S. Patent No. 6,594,785 issued to Gilbertson et al. ("Gilbertson").

VII. ARGUMENT

The Examiner has rejected claims 1-5, 7-17, 19-24, and 26 as unpatentable under 35 U.S.C. § 102(a) as being anticipated by Mathur, and claims 6, 18 and 25 as unpatentable under 35 U.S.C. §103(a) in view of Mathur over Gilbertson.

All of the claims do not stand or fall together. The basis for the separate patentability of the claims is set forth below.

A. Overview of Cited References

1. U.S. Patent No. 5,748,781 issued to Mathur

Mathur discloses a method of controlling memory usage in a computer system having limited physical memory (Abstract). When the memory usage reaches a critical threshold, a user is prompted to select a currently executing application program for termination (col. 4, lines 56-59). Mathur contemplates a computing environment in which application programs compete for available memory (col. 4, limes12-14). Thus, the critical threshold is reached when the application programs overuse their memory resources and, as a result, one or more of the application programs are terminated to release a portion of the memory (Abstract).

Mathur does not disclose any indication of a hardware error. Thus, there is no teaching of the claimed "receiving an indication of a hardware error associated with an application program thread" in the disclosure of Mathur.

2. U.S. Patent No. 6,594,785 issued to Gilbertson

Gilbertson discloses hardware fault isolation and recovery. Gilbertson discloses a multiprocessor system that isolates faults within a failing partition (e.g., a processor) and prevents the faults from creating a failure in a non-failing partition (col. 1, lines 46-50). The faults are isolated based on the physical partition with which a hardware error is associated. The faults are isolated by poisoning specific memory locations.

Gilbertson also fails to disclose "receiving an indication of a hardware error associated with an application program thread." Gilbertson is completely silent on the concept of "a hardware error associated with an application program thread."

B. Claims Rejected Under 35 U.S.C. § 102

1. Claims 1-5, 7-17, 19-24, and 26

Claims 1-5, 7-17, 19-24, and 26 stand rejected under 35 U.S.C. § 102(a) as being anticipated by U.S. Patent No. 6,938,254 issued to Mathur. Appellant submits that Mathur does not teach each of the elements of these claims.

Mathur does not disclose a hardware error in association with an application program thread. Rather, Mathur discloses a method of controlling memory usage in a computer system having limited physical memory (Abstract). When the memory usage reaches a critical threshold, a user is prompted to select a currently executing application program for termination (col. 4, lines 56-59).

In the Final Office Action on page 8, the Examiner indicates that an error occurs when memory usage reaches the threshold or critical memory usage. The Examiner further indicates that "since this error affects memory which is hardware, then Mathur discloses a hardware error."

A hardware error, as generally known in the computer art, is an "error resulting from a malfunction of some physical component of the computer" (see, e.g., the definition provided by wordnet.princeton.edu). Critical memory usage is not an error caused by malfunction of a physical component of the computer. Critical memory usage may affect the use of memory, but is not related to hardware malfunction.

Applying the Examiner's analysis of the "hardware error," even a typical software error that affects memory allocation would be considered as a hardware error. For example, the malloc subroutine in the C programming language's standard library performs dynamic memory allocation (see, e.g., "The C programming language" by

Kernighan and Ritchie, Prentice-Hall, 1978). When the memory allocated by malloc exceeds the available memory size, an error occurs in the C program that calls the malloc subroutine. This malloc error is a software error because it is caused by unsuccessful execution of software code. However, applying the Examiner's analysis, a malloc error would be considered as a hardware error, because it affects the memory in the same way as the overused memory of Mathur. Thus, the above example shows that the Examiner's assertion simply contradicts the common meaning of a hardware error. Thus, the Examiner has made a clear factual error in the rejection, at least with respect to the claimed "hardware error."

For the foregoing reasons, Mathur does not teach each of the elements of claim 1. Claims 2-5 depend from claim 1 and incorporate the limitations thereof. Thus, for at least the reasons mentioned above in regard to claim 1, Mathur does not teach each of the elements of claims 2-5. Analogous discussions apply to independent claims 8, 13, and 20 and their dependent claims, namely, claims 7, 9-12, 14-17, 19, 21-24, and 26.

Thus, for at least the foregoing reasons, Appellant requests the Board to overturn the anticipation rejection of claims 1-5, 7-17, 19-24, and 26.

C. Claims Rejected Under 35 U.S.C. § 103(a) in view of Mathur over Gilbertson

1. Claims 6, 18, and 25

To establish a *prima facie* case of obviousness, the relied upon references must teach or suggest every limitation of the claim such that the invention as a whole would have been obvious at the time the invention was made to one skilled in the art. Claims 6, 18 and, 25 depend from claims 1, 13, and 20 and incorporate the limitations thereof. Thus, for at least the reasons mentioned above in regard to claim 1, Mathur does not teach or suggest each of the elements of these dependent claims.

Claim 6 modifies claim 1 to further include receiving information of a poisoned data address associated with the hardware error. The Examiner relies on Gilbertson for disclosing the claimed receiving information of the poisoned data address associated with the hardware error. However, Gilbertson does not cure the deficiency of Mathur, which fails to disclose "receiving an indication of a hardware error associated with an application program thread" recited in base claim 1. Gilbertson discloses a multiprocessor system that isolates faults within a failing partition (e.g., a processor) and prevents the faults from creating a failure in a non-failing partition (col. 1, lines 46-50). However, Gilbertson does not teach or suggest the concept of "a hardware error associated with an application program thread." The Examiner has not identified and Applicant has been unable to discern any part of Gilbertson that discloses fault isolation based on the application program thread with which a hardware error is associated. Rather, Gilbertson isolates faults based on the physical partition with which a hardware error is associated. Thus, Mathur in view of Gilbertson does not teach or suggest each of the elements of claim 1 and its dependent claim 6.

Moreover, there is no motivation to combine Mathur with Gilbertson. Mathur discloses a method for reducing memory usage by terminating at least one application program. Thus, the method of Mathur handles software errors by terminating software programs. Gilbertson discloses fault handling by poisoning specific memory locations. Thus, the method of Gilbertson handles hardware errors by invalidating faulty hardware. The two references propose distinctly different solutions for solving distinctly different technical problems. Thus, a skilled person would not be motivated to combine the references. See MPEP § 2143.01 I. The Prior Art Must Suggest the Desirability of the Claimed Invention. Thus, the proposed combination is inapposite.

Thus, claim 6, by incorporating the limitations of independent claim 1, is non-obvious over the cited references. Analogous discussions apply to dependent claims 18 and 25, which incorporate the limitations of independent claims 13 and 20, respectively. Accordingly, Appellant requests the Board to overturn the obviousness rejection of claims 6, 18, and 25.

Based on the foregoing, the Board should **overturn** the rejection of all pending claims and hold that all of the claims currently pending in the application under review are allowable.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: January 18, 2007

Tong J. Lee, Reg. No. 48,582

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025

(310) 207-3800

CERTIFICATE OF MAILING

I hereby certify that the correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to:

Mail Stop Appeal Brief - Patents
Commissioner for Patents
Alexandria, VA 22313-1450

Amber D. Saunders 1/18/2007

Amber D. Saunders Date

VIII. CLAIMS APPENDIX

The claims involved in this appeal are presented below.

1. (Original) A method of terminating an affected application program thread, comprising:

receiving an indication of a hardware error associated with an application program thread;

determining the application program thread to be in a user operation mode; and terminating the application program.

2. (Original) The method of claim 1, wherein the terminating the application program further comprises:

determining the hardware error is a memory read error, the memory read error being associated with the application program thread.

3. (Original) The method of claim 2, further comprising:

determining the memory read error is successfully contained.

4. (Original) The method of claim 3, further comprising:

receiving information of whether the memory read error is contained.

5. (Original) The method of claim 2, further comprising:

receiving information of whether the hardware error occurred on a memory read.

6. (Original) The method of claim 1, further comprising:

receiving information of a poisoned data address associated with the hardware error.

7. (Original) The method of claim 1, further comprising:

confirming one or more registers associated with the application program thread are consumed.

8. (Previously Presented) A system comprising:

a processor to perform an instruction from an operating system; and

a memory component to provide machine error information to the operating system, the machine error information to include an operation mode of the affected application program, the operating system to terminate the affected application program thread upon determining the affected application program to be within a user operation mode.

9. (Original) The system of claim 8, wherein the processor is to receive an instruction from the operating system to terminate the affected application program thread upon determining a memory read error has occurred.

10. (Original) The system of claim 9, wherein the processor is to receive an instruction from the operating system to terminate the affected application program thread upon determining the memory read error is contained.

11. (Original) The system of claim 9, wherein the operating system is to check to the machine error information message to determine whether the memory read error occurred.

12. (Original) The system of claim 11, wherein the operating system is to check the machine error information message to determine whether the memory read error is contained.

13. (Original) A machine-accessible medium that provides instructions that, if executed by a machine, will cause the machine to perform operations comprising:

receiving an indication of a hardware error associated with an application program thread;

determining the application program thread to be in a user operation mode; and

terminating the application program.

14. (Original) The machine-accessible medium of claim 13, wherein the terminating the application program further comprises:

determining the hardware error is a memory read error, the memory read error being associated with the application program thread.

15. (Original) The machine-accessible medium of claim 14, further comprising:

determining the memory read error is successfully contained.

16. (Original) The machine-accessible medium of claim 15, further comprising:

receiving information of whether the memory read error is contained.

17. (Original) The machine-accessible medium of claim 14, further comprising:

receiving information of whether the hardware error occurred on a memory read.

18. (Original) The machine-accessible medium of claim 13, further comprising:

receiving information of a poisoned data address associated with the hardware error.

19. (Original) The machine-accessible medium of claim 13, further comprising:

confirming one or more registers associated with the application program thread are consumed.

20. (Original) A system comprising:

a means for receiving an indication of a hardware error associated with an application program thread;

a means for determining the application program thread to be in a user operation mode; and

a means for terminating the application program.

21. (Original) The system of claim 20, wherein the means for terminating the application program further comprises:

a means for determining the hardware error is a memory read error, the memory read error being associated with the application program thread.

22. (Original) The system of claim 21, further comprising:

a means for determining the memory read error is successfully contained.

23. (Original) The system of claim 22, further comprising:

a means for receiving information of whether the memory read error is contained.

24. (Original) The system of claim 23, further comprising:

a means for receiving information of whether the hardware error occurred on a memory read.

25. (Original) The system of claim 20, further comprising:

a means for receiving information of a poisoned data address associated with the hardware error.

26. (Original) The system of claim 20, further comprising:

a means for confirming one or more registers associated with the application program thread are consumed.

IX. EVIDENCE APPENDIX

No evidence is submitted with this appeal brief.

X. RELATED PROCEEDINGS APPENDIX

No related proceedings exist.